If you'd like to be able to see the activity on the band around your frequency, but don't have the budget for a high-end transceiver with a built-in panadapter … or you're looking for a bigger display, WA9PYH may have a solution for you: Build your own panadapter, using a cheap Raspberry Pi microcomputer and a mid-sized HDMI display.

Build a Raspberry Pi 3 Panadapter

BY JIM KOCSIS,* WA9PYH

A panadapter is a device connected to your receiver that allows you to see activity around the frequency where your receiver is tuned. In *QST* for February 2017, I described a panadapter using an SDR dongle, laptop with HDSDR, or a smartphone/tablet and a $10 app, but was never very happy with it. I figured a Raspberry Pi computer could work but didn't know how to write the required software. I subsequently found the perfect software described on YouTube and built a dedicated panadapter using the same hardware but with a smaller display. The Pi 3 uses free software called GQRX that was written by Alexandru Csete, OZ9AEC, in Denmark. To see it in action, search GQRX in YouTube. *Photos A* and *B* show the unit in operation and all the component boards/modules.

Several high-end transceivers (ICOM 7300, 7610, Yaesu FT991A) come with a panadapter that appears on the same touchscreen that is used to select operating conditions. Because these self-contained panadapters have to "share" screen space with the other settings, they are quite small. For us "elders" (with less than perfect vision), a larger display would be very helpful. The display here is a 7-inch diagonal HDMI display with 1024 x 600 pixel resolution. The maximum cost for all parts new is $270, which is less than half the cost of a commercial unit. You may not need them all so your cost could be much less. There are only two connections: Power and the IF signal. All parameters are controlled via menus and a Bluetooth mouse. Power is supplied by a 2.5-Amp, 5-Volt wall wart that is included with the Pi 3 shown in the parts list.

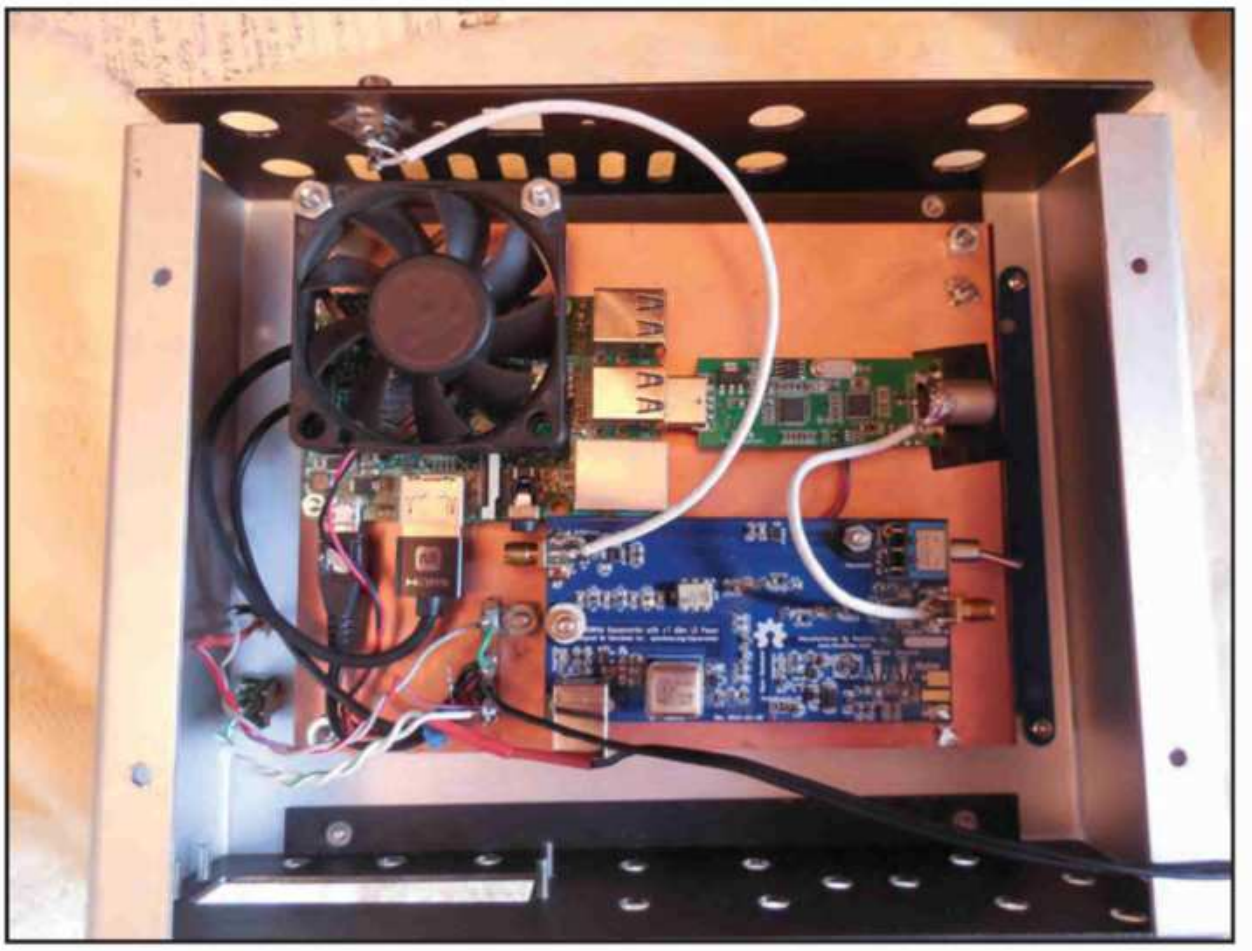

Photo A. All panadapter components mounted inside cabinet.

Controls to Vary the Operating Parameters

The software has many, many parameters you can vary. The parameters of prime interest are: Bandwidth, averaging, waterfall-to-spectrum ratio, FFT size (basically resolution), peak detect, and displayed range of signal amplitude. There is more than enough to keep you learning for a long time, especially FFT (Fast Fourier Transform), which is a subject all in itself. All operation is via a mouse — there are NO KNOBS or touchscreen. A wired mouse and keyboard will be needed initially but after setting up the wireless mouse, it's totally wireless.

The Components and Block Diagram

See Figure 1 to see how the various components are connected. If your IF signal is lower than 53 MHz, you will need an upconverter to shift it into the tuning range of the SDR dongle. If you do not have a buffered IF signal output from your receiver, you will need a buffer amplifier ($45) that has a high impedance input and low impedance

* *email: <sadiekitty@sbcglobal.net>*

Photo B. Panadapter display showing the entire 40-meter band in early evening.

output so the IF stage is not loaded down or detuned. If you have a buffered IF signal you are ready to go.

The longest signal path is: Buffer amplifier inside the rig to upconverter to SDR dongle to Pi 3 USB input to HDMI display.

Software

If you have experience with loading Raspbian into the Pi, using the Terminal editor, preparing an SD card, and Linux you can load all the software needed. Step-by-step instructions are shown in a YouTube video[1] and in a sidebar to this article. If you don't know how or don't feel confident but have a friend who is a Pi expert, enlist his/her help. As an alternative, snail-mail me a 16 GB micro SD card so I can copy my SD card image onto it and mail it back to you. Then all you should have to do is plug it into the Pi and you are ready to go.

If you go it alone using Windows, you will need to use SD Card Formatter and WinDiskImager 32. If you are not completely sure of what you are doing, get help! SD Card Formatter can delete everything on your PCs' hard drive if you select the wrong drive!

Please note that I have made the following changes to GQRX: (1) the program name and folder are "PAN" instead "GQRX-2.6-RPI3-2" (a bit easier to type when I was working on it), (2) the "autostart" file has been modified using the NANO editor to make the program start automatically and (3) the "config.txt" file was modified to change the video to 1024 x 600 pixels to match the 7-inch screen format and disable the screen saver.

Parts List

Raspberry Pi 3, 2.5 A power supply/wall wart, stick-on heatsinks – Amazon, CanaKit Raspberry Pi 3 with 2.5 A Micro USB Power Supply, $43

16 GB class 10 micro SD card, Walmart, etc. $10

NooElec or other brand, RTL2832U & R820T2 Tuner SDR – Amazon, $20

7-inch HDMI display, Elecrow HDMI TFT LCD Display 1024 x 600, Amazon item #646381, $70

Upconverter (only required if your IF is less than 53 MHz), Amazon, NooElec Ham It Up v1.3, $40

Right angle micro USB cable, Amazon, StarTech UUSBHAUB6RA, 6 feet long, USB-A to Right Angle Micro USB-B (only required if your configuration/cabinet space needs a low profile connection for power to the display), $5.50

HDMI-to-HDMI 6-inch cable, Amazon, Monoprice 113574 Ultra Slim series High Speed HDMI cable, (only required if your configuration/cabinet needs a low-profile connection between Pi and display, the display comes with a longer and bulkier HDMI cable), $9

Buffer amplifier (only required if your transceiver does not have a buffered IF output signal), DX Engineering model DXE-Z10000U-KIT (assembled), $45

12 VDC, 2-1/4-inch square fan (junk box, from a PC, etc)

6- x 9-inch copperclad board (used in making PC boards)

Bluetooth mouse (optional) available at Walmart, etc. $20

Terminal strip, 3 terminals

Miscellaneous mounting hardware for 6- x 9-inch board

Metal case to house all components

Table 1. Parts list including sources and prices for some components. You may have other sources or have some of these parts in your junkbox.

Connection to the Rig's IF Signal

If your transceiver provides an IF signal output at an accessory connector, check to see if it is buffered by connecting a 50-ohm resistor from that pin to ground. If there is no degradation in signal, no buffer is required. If the signal does degrade, you'll need a buffer amplifier (see below). If your rig does not provide an IF signal, you will have to tap into the proper point. The website <https://tinyurl.com/zvqn24b> has links to information for many popular rigs.

Scroll down to the first table to find your IF frequency. Dave Powis, G4HUP, the ham who made up this page also made buffer boards but is a Silent Key so his boards are no longer available. Suitable boards can be purchased from DX Engineering as shown in the parts list.

After finding your rig's IF frequency, scroll down to a second table. Links there show where and how to connect to the proper point. PLEASE NOTE: You cannot just tap into the indicated point. A buffer amplifier is required. A suitable buffer amplifier is available from DX Engineering. My rig is a Ten-Tec Eagle and has the 9-MHz IF buffered signal available at the AUX 1 connector on back.

Warnings

Several precautions must be taken in building this project:

- Anytime you handle the boards, use a static wrist strap to ensure you are grounded.
- You MUST install this unit inside a metal case. If not, transmitted RF from your antennas will make it hang up.

• The boards are somewhat delicate and the display should be treated with utmost care.

• I cannot assume any responsibility when you're working inside your rig. I looked over the "how to connect" notes and they appear very complete and accurate.

Construction

RF signal connections between the SDR dongle and upconverter boards are made using small diameter coax so you don't have to buy extra cables; plus it saves space. The shield and center conductor are connected at both ends and are soldered directly to the boards (no need to use the connectors on the boards — see *Photo C*). Power for the upconverter is "stolen" from the SDR dongle (see *Figure 2*).

Adequate-sized wiring must be used for the 5-volt power to the Pi 3 and display. For the Pi 3, I cut the micro USB cable from the supplied wall wart to a length of 6 inches. The red and black wires are +5 volts and ground, respectively. Cut the right-angle micro USB cable for power to the display to 6 inches. The white and green wires aren't needed. The red and black are +5 volt and ground. The SDR dongle is plugged directly into the Pi.

Grounding must be done as shown in *Figure 1* (see note 3 on the figure) or noise from the various boards might mask weaker signals, appear as broadband noise or cause carriers to appear that never change. I spent many hours getting the grounding scheme right — just follow *Figure 1* and your unit will work fine. The heat sink on the CPU chip of the Pi 3 gets quite hot so I placed a cooling fan over the heat sink such that half the fan is aimed at the heat sink. The other half directs air through two holes in the copper board to cool the back of the display. I used a 5/8-inch chassis punch to make the two holes (*Photo D*). A large drill can also be used.

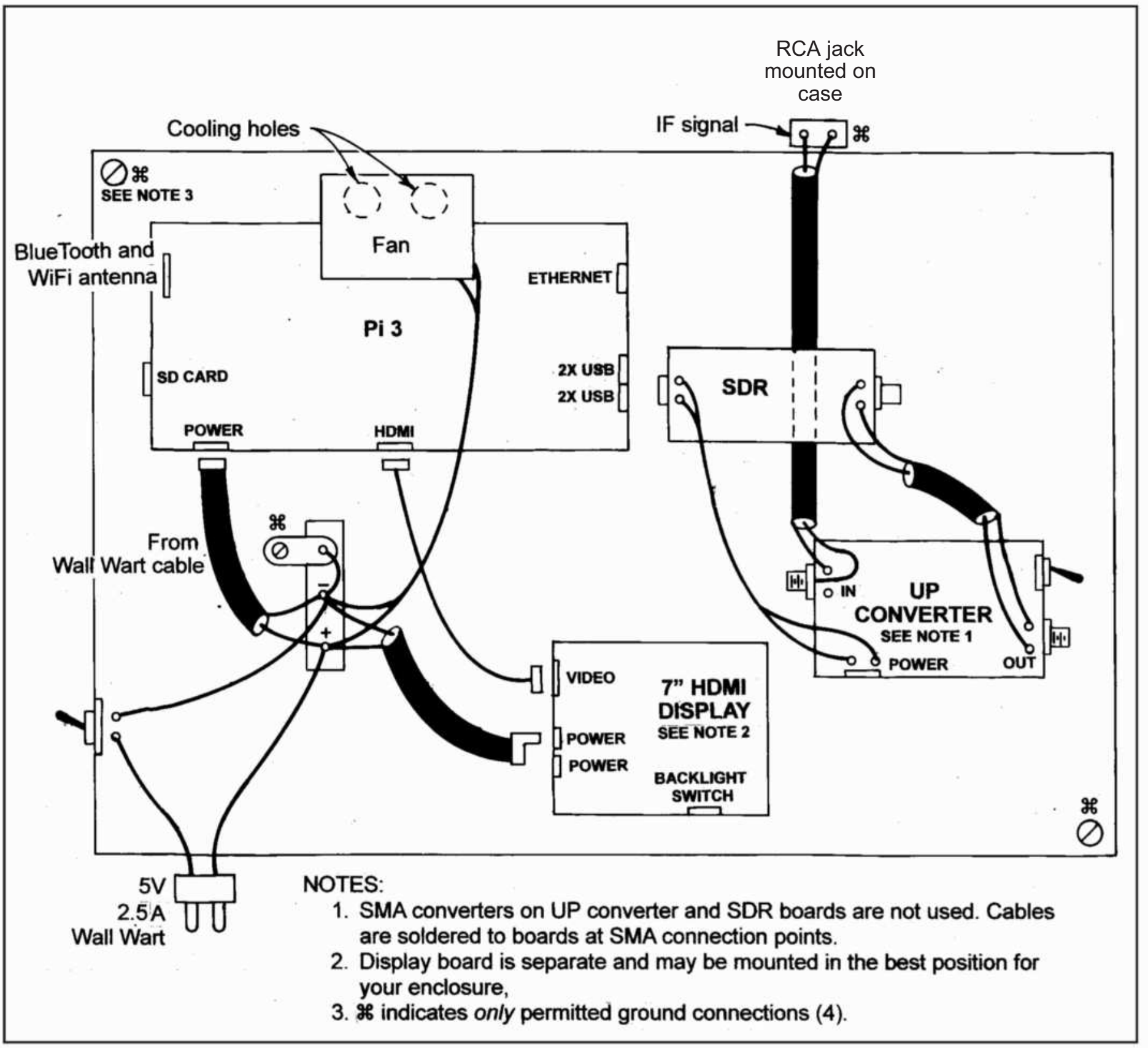

Figure 1. Block diagram of the Raspberry Pi panadapter. Note that the HDMI display module is not necessarily mounted to the same board as the rest of the modules. Arrange your setup in the best way to fit your chosen enclosure, which must be metal.

Cutting the rectangular hole for the display took the most time. I used a saber saw and then filed it as needed. Protect the surface with duct tape. After cutting the rectangular hole, lay the board display-side down so it fits into the hole. Use a pencil to mark the mounting hole locations. Use a 0.140-inch drill to allow for some error in hole location. You will have to supply metric screws to mount the standoffs supplied with the display.

Startup and Running the Program

Bootup takes 35 seconds. Apply power to the unit and you should see the color "splash screen" followed by four raspberry symbols and a lot of text scrolling up the display. If the display doesn't light up, make sure the small slide switch on the back labeled "backlight" is set to "ON." On the upconverter, there is a toggle switch that must be set towards the SMA connector.

To begin use, set the frequency to 0.000 000 MHz at the upper left using the mouse. Using the left mouse button, click on the lower part of each digit to reduce the value and the upper part to increase it. Click on "View" and select "Main Toolbar." Click on the small triangle below "File" to start the program. Click on "View" and select "FFT Settings." Click on "View" again and select "Input Controls."

"Input Controls" is where you will enter the offset frequency into the "LNB LO" window if you use an upconverter — the value is the LO frequency plus your IF. In my case the IF is 9,000,000 (Hz) and the upconverter LO is 100,000,000 (Hz). Precede this entry with a negative sign. My entry is -109,000,000. If you are not using an upconverter, just enter your IF frequency in Hz, preceded by a minus sign. Initially set the LNA gain to 40 dB and the IF gain to 35 dB. By accident, I set both gains to 40 dB and saw signals going both directions across the spectrum as I varied the frequency. This was a case of overload. This effect was especially bad on 40 meters at night when powerful shortwave broadcast stations "pin" the S-meter. The range of the SDR theoretically is about 50 dB since the A/D is an 8-bit device. My good friend and engineer Sam, G4TUH, explained it to me mathematically: An 8-bit A/D has 256 states, so 20 x Log 256 = 48.1 dB. However, I've seen the range of signals go a bit beyond this to about 60 dB.

"FFT Settings" is where you set the following parameters: dB range of the

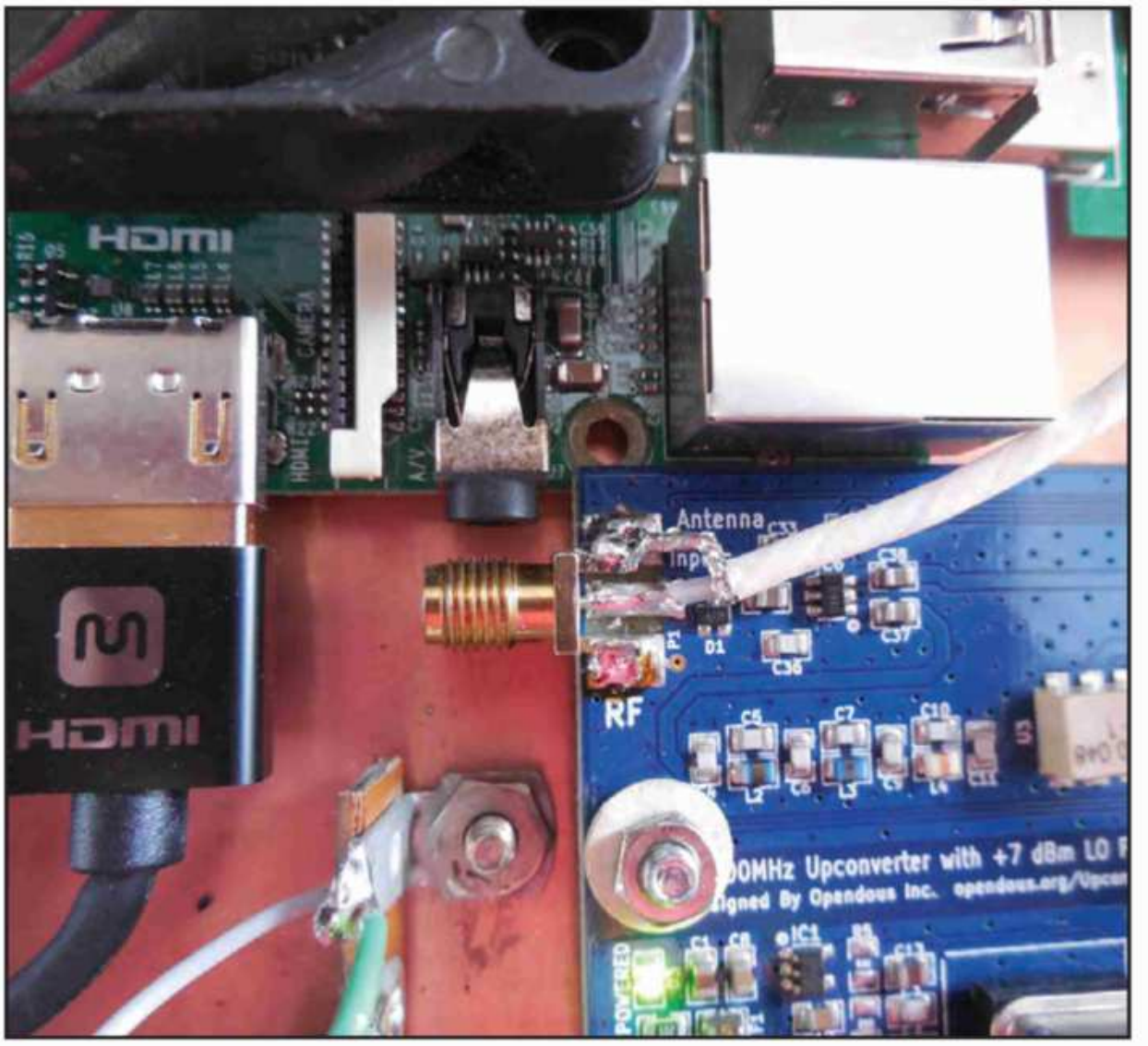

Photo C. Miniature coax cable is used for connections between modules. They are soldered directly to the boards rather than using lossy connectors.

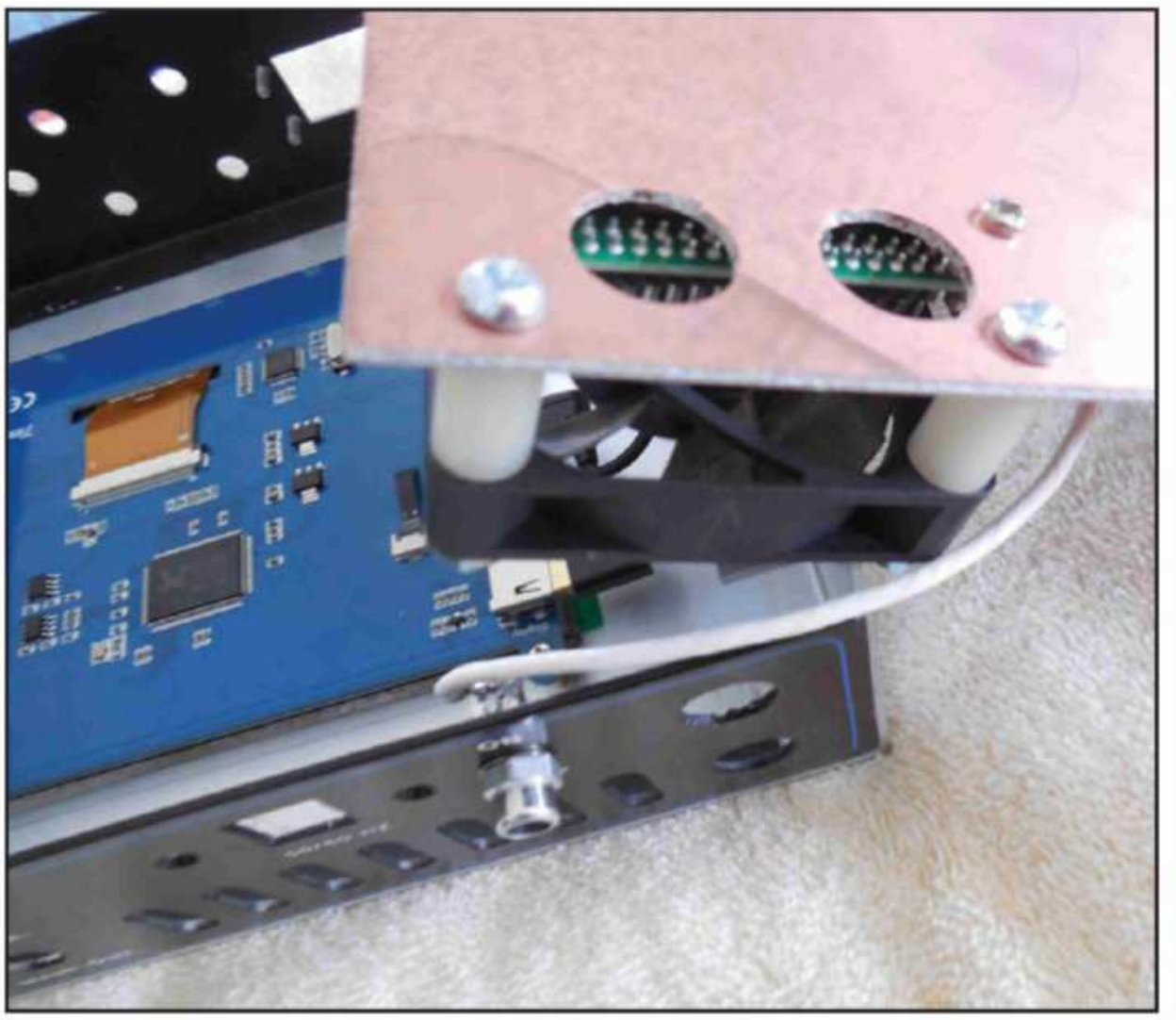

Photo D. A fan was installed to cool the Raspberry Pi's heatsink and to keep the back of the HDMI display cool (through the holes punched in the board). See text for details.

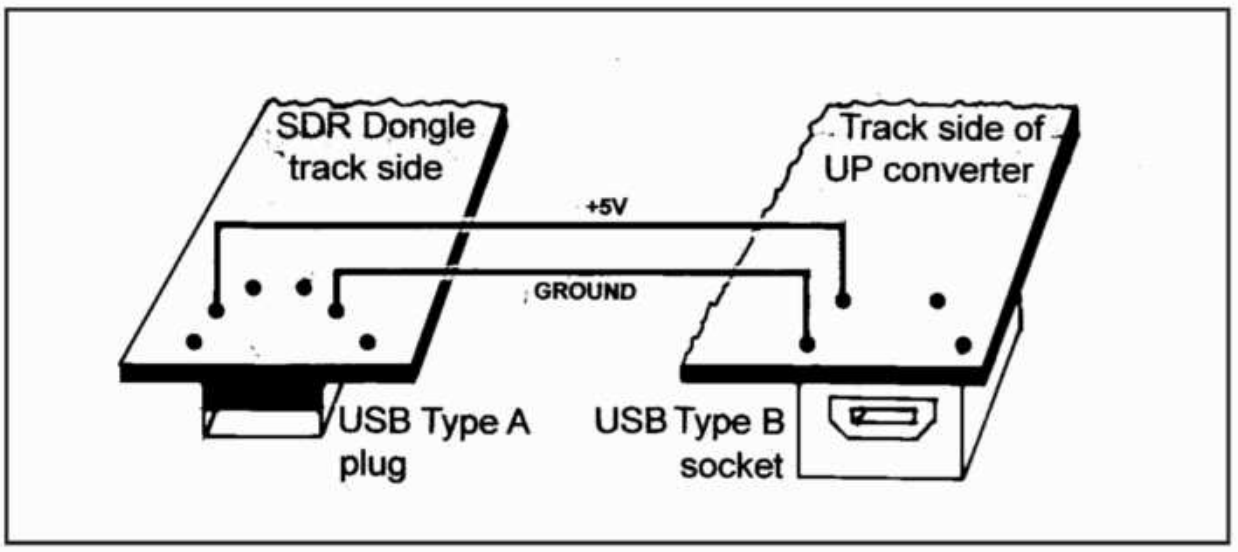

Figure 2. Power connections SDR dongle and upconverter.

spectrum and waterfall, ratio of waterfall-to-spectrum, averaging, bandwidth, peak detect, display colors, and many others. Play with the settings to see the effects of each. You will see CW signals bounce up and down. SSB signals bounce, too, but they are wider. When watching AM signals, you can see the very strong carrier and much weaker sidebands if you zoom way up using the bandwidth slider called "Freq Zoom." The FT8 "band" is especially interesting. Most carriers turn on at 0 and 30 seconds and turn off at 27 & 57 seconds into each minute — some stations' clocks are

Detailed Instructions for Loading the Software

Read Completely Before Proceeding

You need to be somewhat familiar with the Raspberry Pi and Linux or have someone familiar with them to help you. All of these commands are performed in the terminal window (the fourth icon from the left side). You must use a full-sized HDMI display (e.g., a TV with HDMI input), a wired USB keyboard and wired mouse to load the software. Do not connect the 7-inch display until the very end.

• Use the program "SD Card Formatter" to properly prepare the SD card for loading the Pi operating system. SD Card Formatter can be found at <https://tinyurl.com/ybha9q7r>.
• *Make sure you are formatting the SD card and not your hard drive*. This will take about 30 seconds.
• Using your browser, download the operating system from the Raspberry Pi website at <www.raspberrypi.org/downloads>. It is called NOOBS, short for *New Out Of Box Software*.
• Copy the entire folder to the SD card, about 1.26 GB. This should take about 4 minutes.
• Connect the keyboard, mouse, monitor and install the card. Connect the power supply.
• To install the OS, after the Pi starts up, click on the box on the screen. The mouse moves *very* fast and the mouse is white surrounded by black on a white background, so look carefully for it — it really is there!
• Hit Install, then confirm. The install will take approximately 20 minutes. At the "OS installed successfully" message, click OK. Next there will be a lot of scrolling text followed by a highway scene.
• Next, click on the Raspberry symbol (upper left), go to Preferences, Mouse & Keyboard Settings. Choose "Mouse" and set the mouse speed to 1.0 to slow it down. Select "Keyboard" and choose "English (US)" at the top of the list — this is so you can enter any special characters. Click OK and OK.

Next we will connect to your WiFi:

• Place the mouse over two red Xs at the upper right, left click, select your WiFi, enter your password, click OK.
• Open the LX Terminal (the black box at the upper left) — you will be using this from now on. Click on Edit, Preferences, click on "MonoSpace 10," select text size 18, click OK, OK.
• Follow the YouTube video referenced in the article (see note 1) at 4 minutes, 0 seconds. Stop at 4:47 where you *would be* changing to a folder called "gqrx-2.6-rpi3-2". We will now change the folder name to "pan" — much easier to type in.

• Change the folder name by entering: **mv grqx-2.6-rpi3-2 pan** (note that there are spaces between "mv" and "gqrx" and between "3-2" and "pan")
• Enter: **cd pan**

Follow the YouTube video at 4:52

• Enter: **./setup_gqrx.sh** (That is a period "." followed by a forward slash, etc.), plug in the SDR/RTL and upconverter (if you are using an upconverter, make sure the switch on the upconverter is set to "Passthrough," not "Upconvert" for initial testing)
* Enter: **sh run_gqrx.sh** (GQRX will begin running for the first time!)

A lot of text will scroll up the screen. Wait for a pop-up window that says: "Configure I/O Devices"

• Select "Device," then "RealTek"
• Select Sample Rate 2,400,000; click OK.
• Go to upper right and select full screen (just like in Windows®).
• Click the small white triangle below the word "File" in upper left. GQRX will begin running.
• Connect a short wire (1-2 feet long) to the upconverter input connector.
• Using the numbers above the spectrum display, set the frequency to that of a local FM station.

You should see that station on the spectrum display. If not, under "View" select "Input controls" and increase the two gain settings.

• Select "FFT Settings" below the "View" menu at the top of the screen. Adjust the different sliders to see how they affect the display.
• Watch the rest of the YouTube video.

Now we will:

• Make it run at power up ("AutoRun")
• Turn off the screen saver
• Change the screen resolution to that of the 7-inch HDMI display

not set properly so you can see their carriers start early or end late. I have noticed that FT8 signals are nearly always the strongest signals on the band. An interesting phenomenon is a diagonal line on the waterfall while hearing a carrier zip by. It's as if someone was varying their VFO with the key down.

If you're using an upconverter, the spectrum display will be "backwards," i.e., signals higher in frequency will be to the left of center. This occurs when the mixing scheme uses "high side injection" and I have not found a way to reverse it. If you don't use an upconverter, the display will be "correct" with signals higher in frequency to the right of center.

When you're done using the panadapter, you can just shut off the power. When turning it back on, you will get a pop-up window asking if you want to edit the settings. Click "NO" and then it will bootup as normal.

If you have problems, please feel free to contact me and I will help as much as possible.

Conclusion

Once you get the panadapter set up and working, you probably won't be able to get by without it and you can say "*I built it myself!*" I want to thank my wife, Yvonne, for reviewing this article and providing encouragement during the two-year development and learning cycle.

Notes:

1. Step-by-step instructions for loading the software are in a sidebar to this article and on YouTube at <https://tinyurl.com/y8b9qosy>. You will need to refer to both.

AutoRun

• Choose File, Quit to leave GQRX. In LX Terminal, the prompt should show "pi@raspberrypi:~pan$"

• Enter:

chmod +x run_gqrx.sh

sudo nano ~/.config/lxsession/LSDE-pi/autostart

• Add as the last line:

@lxterminal -e /home/pi/pan/run_gqrx.sh

• Hit CTRL-X, then Y, then "enter"

• At the "pi@raspberrypi:~pan$" prompt, enter:

reboot

The Pi will shut down and restart and GQRX should begin running up to the point where you have to hit the small white triangle.

Turn off Screen Saver, Blanking

• In LX terminal, enter:

sudo nano ~/.config/lxsesson/LXDE-pi/autostart

• Enter these 4 lines after the last line (those are the digit zero, not the letter "O"):

xset s 0 0

xset s noblank

xset s noexpose

xset dpms 0 0 0

• Enter CTRL-X, then Y, then return to save changes.

• Reboot, wait for it to restart and *let it run for 15 minutes without touching the keyboard or mouse*. If you made some sort of mistake, the screen will go blank after approximately 10 minutes.

Change resolution to that of the 7-inch HDMI display

• We are going to modify the config.txt file. In LX Terminal, enter the following:

cd ..

cd ..

cd boot

sudo nano config.txt

• Scroll down to the end (bottom) of the config.txt file and enter these 5 lines as the last lines:

max_usb_current=1

hdmi_group=2

hdmi_mode=1

hdmi_mode=87

hdmi_cvt 1024 600 60 6 0 0 0 (those are all zeroes)

• Hit CTRL-X, Y, then return.

• Reboot the Pi and turn off the power. Disconnect the large monitor and connect the 7-inch HDMI display. Power up the Pi and monitor. You should see the same information on the small display as you saw on the large display.

GQRX will work the same as before. Enjoy! If you have problems, go back and make sure you made all entries correctly. If the small display doesn't work, power down the Pi, reconnect the large display and look at the changes to the config.txt file you made in the section above.

Assuming you've been successful, you should make a copy of the SD card on your PC. This way, if your SD card fails, etc., you can reload the software without going through all those steps. You will be saving what is called an image of the file. If you used a 16 GB card, the image will be 16 GB and will take 16 GB of space on your PC.

Save a copy of the SD card on your PC

• To store the complete SD card with GQRX on it, run Win32DiskImager. It is available at <https://tinyurl.com/k33wxrb>.

• Insert the SD card into your PC and note which letter designation is shown for it (e.g., E:, F:, G:, etc).

• In the explorer window, select a folder location where you want to save your image. Double click it and minimize Explorer.

• Type in a filename for the image file (e.g., GQRX_todays_date)

• Confirm the folder and filename are correct.

• Click READ

• Wait until DONE appears in the lower left. (My setup took 12 minutes for a 16 GB SD card).

To restore an image file to an SD card (if your card is damaged or to make a new copy)

• Format a new SD card using SD Card Formatter. Instructions are at the start of this document.

• Make sure the SD card drive letter (e.g. E:, F:, etc) appears under the word "Device." Just to the left is a small symbol of a folder.

• Click on that symbol and navigate to where your saved image file is located.

• Double click on that file name.

• Click "WRITE"

• Wait until it is finished.